

Proyecto_1

Generado por Doxygen 1.8.10

Sábado, 19 de Septiembre de 2015 05:49:48

Índice general

1	Proyecto 1	1
2	Índice de estructura de datos	3
2.1	Estructura de datos	3
3	Índice de archivos	5
3.1	Lista de archivos	5
4	Documentación de las estructuras de datos	7
4.1	Referencia de la Estructura ins_t	7
4.1.1	Documentación de los campos	7
4.1.1.1	array	7
4.2	Referencia de la Estructura instruction_t	7
4.2.1	Documentación de los campos	7
4.2.1.1	mnemonic	7
4.2.1.2	op1_type	7
4.2.1.3	op1_value	8
4.2.1.4	op2_type	8
4.2.1.5	op2_value	8
4.2.1.6	op3_type	8
4.2.1.7	op3_value	8
5	Documentación de archivos	9
5.1	Referencia del Archivo decoder.c	9
5.1.1	Documentación de los 'defines'	9
5.1.1.1	PC	9
5.1.2	Documentación de las funciones	9
5.1.2.1	countLines(FILE *fp)	9
5.1.2.2	decodeInstruction(instruction_t instruction, uint32_t *dir_reg, char *dir_flags)	9
5.1.2.3	getInstruction(char *instStr)	9
5.1.2.4	readFile(char *filename, ins_t *instructions)	10
5.2	Referencia del Archivo decoder.h	10
5.2.1	Documentación de las funciones	10

5.2.1.1	countLines(FILE *fp)	10
5.2.1.2	decodeInstruction(instruction_t instruction, uint32_t *dir_reg, char *dir_flags)	10
5.2.1.3	getInstruction(char *instStr)	10
5.2.1.4	readFile(char *filename, ins_t *instructions)	10
5.3	Referencia del Archivo imprimir.c	10
5.3.1	Documentación de las funciones	11
5.3.1.1	valor_registro(uint32_t valores[])	11
5.4	Referencia del Archivo imprimir.h	11
5.4.1	Documentación de las funciones	11
5.4.1.1	valor_registro(uint32_t valores[])	11
5.5	Referencia del Archivo instruc_desplazamiento.c	11
5.5.1	Documentación de los 'defines'	13
5.5.1.1	C	13
5.5.1.2	N	13
5.5.1.3	V	13
5.5.1.4	Z	13
5.5.2	Documentación de las funciones	13
5.5.2.1	ASR(uint32_t Rdn, uint32_t Rm, char *dir_flags)	13
5.5.2.2	BIC(uint32_t Rdn, uint32_t Rm, char *dir_flags)	13
5.5.2.3	LSL(uint32_t Rdn, uint32_t Rm, char *dir_flags)	14
5.5.2.4	LSR(uint32_t Rdn, uint32_t Rm, char *dir_flags)	14
5.5.2.5	MVN(uint32_t Rdn, char *dir_flags)	15
5.5.2.6	REV(uint32_t Rdn)	15
5.5.2.7	REVG(uint32_t Rdn)	15
5.5.2.8	REVSH(uint32_t Rdn)	16
5.5.2.9	ROR(uint32_t Rdn, uint32_t Rm, char *dir_flags)	16
5.5.2.10	RSB(uint32_t Rdn, char *dir_flags)	16
5.6	Referencia del Archivo instruc_desplazamiento.h	17
5.6.1	Documentación de las funciones	18
5.6.1.1	ASR(uint32_t Rdn, uint32_t Rm, char *dir_flags)	18
5.6.1.2	BIC(uint32_t Rdn, uint32_t Rm, char *dir_flags)	19
5.6.1.3	LSL(uint32_t Rdn, uint32_t Rm, char *dir_flags)	19
5.6.1.4	LSR(uint32_t Rdn, uint32_t Rm, char *dir_flags)	19
5.6.1.5	MVN(uint32_t Rdn, char *dir_flags)	20
5.6.1.6	REV(uint32_t Rdn)	20
5.6.1.7	REVG(uint32_t Rdn)	21
5.6.1.8	REVSH(uint32_t Rdn)	21
5.6.1.9	ROR(uint32_t Rdn, uint32_t Rm, char *dir_flags)	21
5.6.1.10	RSB(uint32_t Rdn, char *dir_flags)	22
5.7	Referencia del Archivo instrucciones.c	22

5.7.1	Documentación de los 'defines'	24
5.7.1.1	C	24
5.7.1.2	N	24
5.7.1.3	PC	24
5.7.1.4	V	24
5.7.1.5	Z	24
5.7.2	Documentación de las funciones	24
5.7.2.1	ADC(uint32_t Rn, uint32_t Rm, char *dir_flags)	25
5.7.2.2	ADD(uint32_t Rn, uint32_t Rm, char *dir_flags)	25
5.7.2.3	AND(uint32_t Rn, uint32_t Rm, char *dir_flags)	25
5.7.2.4	CMN(uint32_t Rn, uint32_t Rm, char *dir_flags)	26
5.7.2.5	CMP(uint32_t Rn, uint32_t Rm, char *dir_flags)	26
5.7.2.6	EOR(uint32_t Rn, uint32_t Rm, char *dir_flags)	26
5.7.2.7	flag_C(uint32_t Rn, uint32_t Rm, uint32_t Rd, char *dir_flag_C)	27
5.7.2.8	flag_N(uint32_t Rd, char *dir_flag_N)	27
5.7.2.9	flag_V(uint32_t Rn, uint32_t Rm, uint32_t Rd, char *dir_flag_V)	27
5.7.2.10	flag_Z(uint32_t Rd, char *dir_flag_Z)	28
5.7.2.11	flags(uint32_t Rn, uint32_t Rm, uint32_t Rd, char *dir_flags)	28
5.7.2.12	MOV(uint32_t Rn, char *dir_flags)	28
5.7.2.13	MUL(uint32_t Rn, uint32_t Rm, char *dir_flags)	29
5.7.2.14	NOP(uint32_t *dir_reg)	29
5.7.2.15	ORR(uint32_t Rn, uint32_t Rm, char *dir_flags)	29
5.7.2.16	SBC(uint32_t Rn, uint32_t Rm, char *dir_flags)	30
5.7.2.17	SUB(uint32_t Rn, uint32_t Rm, char *dir_flags)	30
5.7.2.18	TST(uint32_t Rn, uint32_t Rm, char *dir_flags)	31
5.8	Referencia del Archivo instrucciones.h	31
5.8.1	Documentación de las funciones	33
5.8.1.1	ADC(uint32_t Rn, uint32_t Rm, char *dir_flags)	33
5.8.1.2	ADD(uint32_t Rn, uint32_t Rm, char *dir_flags)	33
5.8.1.3	AND(uint32_t Rn, uint32_t Rm, char *dir_flags)	34
5.8.1.4	CMN(uint32_t Rn, uint32_t Rm, char *dir_flags)	34
5.8.1.5	CMP(uint32_t Rn, uint32_t Rm, char *dir_flags)	34
5.8.1.6	EOR(uint32_t Rn, uint32_t Rm, char *dir_flags)	35
5.8.1.7	flag_C(uint32_t Rn, uint32_t Rm, uint32_t Rd, char *dir_flag_C)	35
5.8.1.8	flag_N(uint32_t Rd, char *dir_flag_N)	35
5.8.1.9	flag_V(uint32_t Rn, uint32_t Rm, uint32_t Rd, char *dir_flag_V)	36
5.8.1.10	flag_Z(uint32_t Rd, char *dir_flag_Z)	36
5.8.1.11	flags(uint32_t Rn, uint32_t Rm, uint32_t Rd, char *dir_flags)	36
5.8.1.12	MOV(uint32_t Rn, char *dir_flags)	38
5.8.1.13	MUL(uint32_t Rn, uint32_t Rm, char *dir_flags)	38

5.8.1.14	NOP(uint32_t *dir_reg)	39
5.8.1.15	ORR(uint32_t Rn, uint32_t Rm, char *dir_flags)	39
5.8.1.16	SBC(uint32_t Rn, uint32_t Rm, char *dir_flags)	39
5.8.1.17	SUB(uint32_t Rn, uint32_t Rm, char *dir_flags)	40
5.8.1.18	TST(uint32_t Rn, uint32_t Rm, char *dir_flags)	40
5.9	Referencia del Archivo main.c	40
5.9.1	Documentación de los 'defines'	41
5.9.1.1	C	41
5.9.1.2	LR	41
5.9.1.3	N	41
5.9.1.4	PC	41
5.9.1.5	V	41
5.9.1.6	Z	41
5.9.2	Documentación de las funciones	41
5.9.2.1	main(void)	41
5.10	Referencia del Archivo README.md	41
5.11	Referencia del Archivo salto.c	41
5.11.1	Documentación de los 'defines'	43
5.11.1.1	C	43
5.11.1.2	LR	43
5.11.1.3	N	43
5.11.1.4	PC	44
5.11.1.5	V	44
5.11.1.6	Z	44
5.11.2	Documentación de las funciones	44
5.11.2.1	B(uint32_t label, uint32_t *dir_reg)	44
5.11.2.2	BAL(uint32_t label, uint32_t *dir_reg)	44
5.11.2.3	BCC(uint32_t label, uint32_t *dir_reg, char *dir_flags)	44
5.11.2.4	BCS(uint32_t label, uint32_t *dir_reg, char *dir_flags)	45
5.11.2.5	BEQ(uint32_t label, uint32_t *dir_reg, char *dir_flags)	45
5.11.2.6	BGE(uint32_t label, uint32_t *dir_reg, char *dir_flags)	45
5.11.2.7	BGT(uint32_t label, uint32_t *dir_reg, char *dir_flags)	46
5.11.2.8	BHI(uint32_t label, uint32_t *dir_reg, char *dir_flags)	46
5.11.2.9	BL(uint32_t label, uint32_t *dir_reg)	46
5.11.2.10	BLE(uint32_t label, uint32_t *dir_reg, char *dir_flags)	47
5.11.2.11	BLS(uint32_t label, uint32_t *dir_reg, char *dir_flags)	47
5.11.2.12	BLT(uint32_t label, uint32_t *dir_reg, char *dir_flags)	47
5.11.2.13	BMI(uint32_t label, uint32_t *dir_reg, char *dir_flags)	48
5.11.2.14	BNE(uint32_t label, uint32_t *dir_reg, char *dir_flags)	48
5.11.2.15	BPL(uint32_t label, uint32_t *dir_reg, char *dir_flags)	48

5.11.2.16 BVC(uint32_t label, uint32_t *dir_reg, char *dir_flags)	49
5.11.2.17 BVS(uint32_t label, uint32_t *dir_reg, char *dir_flags)	49
5.11.2.18 BX(uint32_t *dir_reg)	49
5.12 Referencia del Archivo salto.h	50
5.12.1 Documentación de las funciones	52
5.12.1.1 B(uint32_t label, uint32_t *dir_reg)	52
5.12.1.2 BAL(uint32_t label, uint32_t *dir_reg)	52
5.12.1.3 BCC(uint32_t label, uint32_t *dir_reg, char *dir_flags)	52
5.12.1.4 BCS(uint32_t label, uint32_t *dir_reg, char *dir_flags)	53
5.12.1.5 BEQ(uint32_t label, uint32_t *dir_reg, char *dir_flags)	53
5.12.1.6 BGE(uint32_t label, uint32_t *dir_reg, char *dir_flags)	53
5.12.1.7 BGT(uint32_t label, uint32_t *dir_reg, char *dir_flags)	54
5.12.1.8 BHI(uint32_t label, uint32_t *dir_reg, char *dir_flags)	54
5.12.1.9 BL(uint32_t label, uint32_t *dir_reg)	54
5.12.1.10 BLE(uint32_t label, uint32_t *dir_reg, char *dir_flags)	55
5.12.1.11 BLS(uint32_t label, uint32_t *dir_reg, char *dir_flags)	55
5.12.1.12 BLT(uint32_t label, uint32_t *dir_reg, char *dir_flags)	55
5.12.1.13 BMI(uint32_t label, uint32_t *dir_reg, char *dir_flags)	56
5.12.1.14 BNE(uint32_t label, uint32_t *dir_reg, char *dir_flags)	56
5.12.1.15 BPL(uint32_t label, uint32_t *dir_reg, char *dir_flags)	56
5.12.1.16 BVC(uint32_t label, uint32_t *dir_reg, char *dir_flags)	57
5.12.1.17 BVS(uint32_t label, uint32_t *dir_reg, char *dir_flags)	57
5.12.1.18 BX(uint32_t *dir_reg)	57
5.13 Referencia del Archivo test.c	58
5.13.1 Documentación de los 'defines'	58
5.13.1.1 C	58
5.13.1.2 N	58
5.13.1.3 V	58
5.13.1.4 Z	58
5.13.2 Documentación de las funciones	58
5.13.2.1 main()	58

Capítulo 1

Proyecto 1

Microprocesadores

John Alejandro Barahona Pineda - (alejo7053) Heyler Stivens Montoya Orjuela - (heylermontoya) Javier Andres Sierra Pineda - (sierra112)

Capítulo 2

Índice de estructura de datos

2.1. Estructura de datos

Lista de estructuras con una breve descripción:

ins_t	7
instruction_t	7

Capítulo 3

Indice de archivos

3.1. Lista de archivos

Lista de todos los archivos con descripciones breves:

decoder.c	9
decoder.h	10
imprimir.c	10
imprimir.h	11
instruc_desplazamiento.c	11
instruc_desplazamiento.h	17
instrucciones.c	22
instrucciones.h	31
main.c	40
salto.c	41
salto.h	50
test.c	58

Capítulo 4

Documentación de las estructuras de datos

4.1. Referencia de la Estructura ins_t

```
#include <decoder.h>
```

Campos de datos

- char ** [array](#)

4.1.1. Documentación de los campos

4.1.1.1. char** array

La documentación para esta estructura fue generada a partir del siguiente fichero:

- [decoder.h](#)

4.2. Referencia de la Estructura instruction_t

```
#include <decoder.h>
```

Campos de datos

- char [mnemonic](#) [10]
- char [op1_type](#)
- char [op2_type](#)
- char [op3_type](#)
- uint32_t [op1_value](#)
- uint32_t [op2_value](#)
- uint32_t [op3_value](#)

4.2.1. Documentación de los campos

4.2.1.1. char mnemonic[10]

4.2.1.2. char op1_type

4.2.1.3. uint32_t op1_value

4.2.1.4. char op2_type

4.2.1.5. uint32_t op2_value

4.2.1.6. char op3_type

4.2.1.7. uint32_t op3_value

La documentación para esta estructura fue generada a partir del siguiente fichero:

- [decoder.h](#)

Capítulo 5

Documentación de archivos

5.1. Referencia del Archivo decoder.c

```
#include "decoder.h"
```

'defines'

- #define PC 15

Funciones

- void decodeInstruction (instruction_t instruction, uint32_t *dir_reg, char *dir_flags)
- instruction_t getInstruction (char *instStr)
Obtiene la instrucción separada por partes.
- int readFile (char *filename, ins_t *instructions)
- int countLines (FILE *fp)

5.1.1. Documentación de los 'defines'

5.1.1.1. #define PC 15

5.1.2. Documentación de las funciones

5.1.2.1. int countLines (FILE * fp)

5.1.2.2. void decodeInstruction (instruction_t instruction, uint32_t * dir_reg, char * dir_flags)

5.1.2.3. instruction_t getInstruction (char * instStr)

Obtiene la instrucción separada por partes.

Parámetros

instStr	cadena que contiene la instrucción.
---------	-------------------------------------

Devuelve

instruction_t la instrucción separada por partes.

5.1.2.4. `int readFile (char * filename, ins_t * instructions)`

5.2. Referencia del Archivo decoder.h

```
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <stdint.h>
#include "instrucciones.h"
#include "instruc_desplazamiento.h"
#include "salto.h"
#include <curses.h>
```

Estructuras de datos

- struct [ins_t](#)
- struct [instruction_t](#)

Funciones

- void [decodeInstruction](#) ([instruction_t](#) instruction, uint32_t *dir_reg, char *dir_flags)
- [instruction_t getInstruction](#) (char *instStr)
Obtiene la instrucción separada por partes.
- int [readFile](#) (char *filename, [ins_t](#) *instructions)
- int [countLines](#) (FILE *fp)

5.2.1. Documentación de las funciones

5.2.1.1. `int countLines (FILE * fp)`

5.2.1.2. `void decodeInstruction (instruction_t instruction, uint32_t * dir_reg, char * dir_flags)`

5.2.1.3. `instruction_t getInstruction (char * instStr)`

Obtiene la instrucción separada por partes.

Parámetros

<i>instStr</i>	cadena que contiene la instrucción.
----------------	-------------------------------------

Devuelve

[instruction_t](#) la instrucción separada por partes.

5.2.1.4. `int readFile (char * filename, ins_t * instructions)`

5.3. Referencia del Archivo imprimir.c

```
#include "imprimir.h"
```

Funciones

- void `valor_registro` (uint32_t valores[])

`valor_registro`

prototipo de la funcion valor_registro que muestra el valor que tienen los registros

5.3.1. Documentación de las funciones

5.3.1.1. void `valor_registro` (uint32_t valores[])

`valor_registro`

prototipo de la funcion `valor_registro` que muestra el valor que tienen los registros

5.4. Referencia del Archivo imprimir.h

```
#include <stdio.h>
#include <stdint.h>
#include <stdlib.h>
#include <curses.h>
```

Funciones

- void `valor_registro` (uint32_t valores[])

`valor_registro`

prototipo de la funcion valor_registro que muestra el valor que tienen los registros

5.4.1. Documentación de las funciones

5.4.1.1. void `valor_registro` (uint32_t valores[])

`valor_registro`

prototipo de la funcion `valor_registro` que muestra el valor que tienen los registros

5.5. Referencia del Archivo instruc_desplazamiento.c

```
#include "instruc_desplazamiento.h"
```

'defines'

- #define **N** 0
- #define **Z** 1
- #define **C** 2
- #define **V** 3

Funciones

- uint32_t **LSL** (uint32_t Rdn, uint32_t Rm, char *dir_flags)
LSL

prototipo de la funcion de desplazamiento LSL que va a operar los registros.

- uint32_t **LSR** (uint32_t Rdn, uint32_t Rm, char *dir_flags)
LSR

prototipo de la funcion de desplazamiento LSR que va a operar los registros

- uint32_t **ROR** (uint32_t Rdn, uint32_t Rm, char *dir_flags)
ROR

prototipo de la funcion de desplazamiento ROR que va a operar los registros

- uint32_t **ASR** (uint32_t Rdn, uint32_t Rm, char *dir_flags)
ASR

prototipo de la funcion de desplazamiento ASR que va a operar los registros

- uint32_t **BIC** (uint32_t Rdn, uint32_t Rm, char *dir_flags)
BIC

prototipo de la funcion de desplazamiento BIC que va a operar los registros

- uint32_t **MVN** (uint32_t Rdn, char *dir_flags)
MVN

prototipo de la funcion de desplazamiento MVN que va a operar los registros

- uint32_t **RSB** (uint32_t Rdn, char *dir_flags)
RSB

prototipo de la funcion de desplazamiento RSB que va a operar los registros

- uint32_t **REV** (uint32_t Rdn)
REV

prototipo de la funcion de desplazamiento REV que va a operar los registros

- `uint32_t REVG (uint32_t Rdn)`

REVG

prototipo de la funcion de desplazamiento REV que va a operar los registros

- `uint32_t REVSH (uint32_t Rdn)`

REVSH

prototipo de la funcion de desplazamiento REV que va a operar los registros

5.5.1. Documentación de los 'defines'

5.5.1.1. `#define C 2`

5.5.1.2. `#define N 0`

5.5.1.3. `#define V 3`

5.5.1.4. `#define Z 1`

5.5.2. Documentación de las funciones

5.5.2.1. `uint32_t ASR (uint32_t Rdn, uint32_t Rm, char * dir_flags)`

ASR

prototipo de la funcion de desplazamiento ASR que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion ASR

5.5.2.2. `uint32_t BIC (uint32_t Rdn, uint32_t Rm, char * dir_flags)`

BIC

prototipo de la funcion de desplazamiento BIC que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion BIC

5.5.2.3. uint32_t LSL (uint32_t *Rdn*, uint32_t *Rm*, char * *dir_flags*)**LSL**

prototipo de la funcion de desplazamiento LSL que va a operar los registros.

Parámetros

<i>Rdn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion LSL

5.5.2.4. uint32_t LSR (uint32_t *Rdn*, uint32_t *Rm*, char * *dir_flags*)**LSR**

prototipo de la funcion de desplazamiento LSR que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
<i>Rm</i>	operando ingresado

<i>dir_flags</i>	puntero que tiene la direccion de las banderas
------------------	------------------------------------------------

Devuelve

retorna el resultado de la operacion LSR

5.5.2.5. uint32_t MVN (uint32_t Rdn, char * dir_flags)**MVN**

prototipo de la funcion de desplazamiento MVN que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion MVN

5.5.2.6. uint32_t REV (uint32_t Rdn)**REV**

prototipo de la funcion de desplazamiento REV que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
------------	--------------------

Devuelve

retorna el resultado de la operacion REV

5.5.2.7. uint32_t REVG (uint32_t Rdn)**REVG**

prototipo de la funcion de desplazamiento REV que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
------------	--------------------

Devuelve

retorna el resultado de la operacion REVG

5.5.2.8. uint32_t REVSH (uint32_t *Rdn*)**REVSH**

prototipo de la funcion de desplazamiento REV que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
------------	--------------------

Devuelve

retorna el resultado de la operacion REVSH

5.5.2.9. uint32_t ROR (uint32_t *Rdn*, uint32_t *Rm*, char * *dir_flags*)**ROR**

prototipo de la funcion de desplazamiento ROR que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion ROR

5.5.2.10. uint32_t RSB (uint32_t *Rdn*, char * *dir_flags*)**RSB**

prototipo de la funcion de desplazamiento RSB que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion RSB

5.6. Referencia del Archivo `instruc_desplazamiento.h`

```
#include <stdio.h>
#include <stdint.h>
#include <stdlib.h>
#include "instrucciones.h"
```

Funciones

- uint32_t **LSL** (uint32_t Rdn, uint32_t Rm, char *dir_flags)

LSL

prototipo de la funcion de desplazamiento LSL que va a operar los registros.

- uint32_t **LSR** (uint32_t Rdn, uint32_t Rm, char *dir_flags)

LSR

prototipo de la funcion de desplazamiento LSR que va a operar los registros

- uint32_t **ROR** (uint32_t Rdn, uint32_t Rm, char *dir_flags)

ROR

prototipo de la funcion de desplazamiento ROR que va a operar los registros

- uint32_t **ASR** (uint32_t Rdn, uint32_t Rm, char *dir_flags)

ASR

prototipo de la funcion de desplazamiento ASR que va a operar los registros

- uint32_t **BIC** (uint32_t Rdn, uint32_t Rm, char *dir_flags)

BIC

prototipo de la funcion de desplazamiento BIC que va a operar los registros

- uint32_t **MVN** (uint32_t Rdn, char *dir_flags)

MVN

prototipo de la funcion de desplazamiento MVN que va a operar los registros

- uint32_t **RSB** (uint32_t Rdn, char *dir_flags)

RSB

prototipo de la funcion de desplazamiento RSB que va a operar los registros

- uint32_t **REV** (uint32_t Rdn)

REV

prototipo de la funcion de desplazamiento REV que va a operar los registros

- uint32_t **REVG** (uint32_t Rdn)

REVG

prototipo de la funcion de desplazamiento REV que va a operar los registros

- uint32_t **REVSH** (uint32_t Rdn)

REVSH

prototipo de la funcion de desplazamiento REV que va a operar los registros

5.6.1. Documentación de las funciones

5.6.1.1. uint32_t ASR (uint32_t Rdn, uint32_t Rm, char * dir_flags)

ASR

prototipo de la funcion de desplazamiento ASR que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion ASR

5.6.1.2. `uint32_t BIC (uint32_t Rdn, uint32_t Rm, char * dir_flags)`**BIC**

prototipo de la funcion de desplazamiento BIC que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion BIC

5.6.1.3. `uint32_t LSL (uint32_t Rdn, uint32_t Rm, char * dir_flags)`**LSL**

prototipo de la funcion de desplazamiento LSL que va a operar los registros.

Parámetros

<i>Rdn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion LSL

5.6.1.4. `uint32_t LSR (uint32_t Rdn, uint32_t Rm, char * dir_flags)`**LSR**

prototipo de la funcion de desplazamiento LSR que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion LSR

5.6.1.5. uint32_t MVN (uint32_t *Rdn*, char * *dir_flags*)**MVN**

prototipo de la funcion de desplazamiento MVN que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion MVN

5.6.1.6. uint32_t REV (uint32_t *Rdn*)**REV**

prototipo de la funcion de desplazamiento REV que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
------------	--------------------

Devuelve

retorna el resultado de la operacion REV

5.6.1.7. `uint32_t REVG (uint32_t Rdn)`**REVG**

prototipo de la funcion de desplazamiento REV que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
------------	--------------------

Devuelve

retorna el resultado de la operacion REVG

5.6.1.8. `uint32_t REVSH (uint32_t Rdn)`**REVSH**

prototipo de la funcion de desplazamiento REV que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
------------	--------------------

Devuelve

retorna el resultado de la operacion REVSH

5.6.1.9. `uint32_t ROR (uint32_t Rdn, uint32_t Rm, char * dir_flags)`**ROR**

prototipo de la funcion de desplazamiento ROR que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
------------	--------------------

<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion ROR

5.6.1.10. uint32_t RSB (uint32_t Rdn, char * dir_flags)

RSB

prototipo de la funcion de desplazamiento RSB que va a operar los registros

Parámetros

<i>Rdn</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion RSB

5.7. Referencia del Archivo instrucciones.c

```
#include "instrucciones.h"
```

'defines'

- #define **N** 0
- #define **Z** 1
- #define **C** 2
- #define **V** 3
- #define **PC** 15

Funciones

- void **flags** (uint32_t Rn, uint32_t Rm, uint32_t Rd, char *dir_flags)
flags

prototipo de la funcion flags que va a operar los registros de las banderas

- void **flag_N** (uint32_t Rd, char *dir_flag_N)

flag_N

prototipo de la funcion flag_N que va a operar el registro de la bandera N

- void **flag_Z** (uint32_t Rd, char *dir_flag_Z)

flag_Z

prototipo de la funcion flag_Z que va a operar el registro de la bandera Z

- void **flag_C** (uint32_t Rn, uint32_t Rm, uint32_t Rd, char *dir_flag_C)

flag_C

prototipo de la funcion flag_C que va a operar el registro de la bandera C

- void **flag_V** (uint32_t Rn, uint32_t Rm, uint32_t Rd, char *dir_flag_V)

flag_V

prototipo de la funcion flag_V que va a operar el registro de la bandera V

- void **CMN** (uint32_t Rn, uint32_t Rm, char *dir_flags)

CMN

prototipo de la funcion CMN que va a operar los registros

- void **CMP** (uint32_t Rn, uint32_t Rm, char *dir_flags)

CMP

prototipo de la funcion CMP que va a operar los registros

- uint32_t **MUL** (uint32_t Rn, uint32_t Rm, char *dir_flags)

MUL

prototipo de la funcion MUL que va a operar los registros

- void **TST** (uint32_t Rn, uint32_t Rm, char *dir_flags)

TST

prototipo de la funcion TST que va a operar los registros

- uint32_t **ADD** (uint32_t Rn, uint32_t Rm, char *dir_flags)

ADD

prototipo de la funcion ADD que va a operar los registros

- uint32_t **ADC** (uint32_t Rn, uint32_t Rm, char *dir_flags)

ADC

prototipo de la funcion ADC que va a operar los registros

- uint32_t **SUB** (uint32_t Rn, uint32_t Rm, char *dir_flags)
- SUB**

prototipo de la funcion SUB que va a operar los registros

- uint32_t **SBC** (uint32_t Rn, uint32_t Rm, char *dir_flags)
- SBC**

prototipo de la funcion SBC que va a operar los registros

- uint32_t **MOV** (uint32_t Rn, char *dir_flags)
- MOV**

prototipo de la funcion MOV que va a operar los registros

- uint32_t **AND** (uint32_t Rn, uint32_t Rm, char *dir_flags)
- AND**

prototipo de la funcion AND que va a operar los registros

- uint32_t **EOR** (uint32_t Rn, uint32_t Rm, char *dir_flags)
- EOR**

prototipo de la funcion EOR que va a operar los registros

- uint32_t **ORR** (uint32_t Rn, uint32_t Rm, char *dir_flags)
- ORR**

prototipo de la funcion ORR que va a operar los registros

- void **NOP** (uint32_t *dir_reg)
- NOP**

prototipo de la funcion NOP que va a operar los registros

5.7.1. Documentación de los 'defines'

5.7.1.1. **#define C 2**

5.7.1.2. **#define N 0**

5.7.1.3. **#define PC 15**

5.7.1.4. **#define V 3**

5.7.1.5. **#define Z 1**

5.7.2. Documentación de las funciones

5.7.2.1. uint32_t ADC (uint32_t *Rn*, uint32_t *Rm*, char * *dir_flags*)

ADC

prototipo de la funcion ADC que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresaaado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la ADC

5.7.2.2. uint32_t ADD (uint32_t *Rn*, uint32_t *Rm*, char * *dir_flags*)

ADD

prototipo de la funcion ADD que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresaaado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la ADD

5.7.2.3. uint32_t AND (uint32_t *Rn*, uint32_t *Rm*, char * *dir_flags*)

AND

prototipo de la funcion AND que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion AND

5.7.2.4. void CMN (uint32_t *Rn*, uint32_t *Rm*, char * *dir_flags*)

CMN

prototipo de la funcion CMN que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.7.2.5. void CMP (uint32_t *Rn*, uint32_t *Rm*, char * *dir_flags*)

CMP

prototipo de la funcion CMP que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.7.2.6. uint32_t EOR (uint32_t *Rn*, uint32_t *Rm*, char * *dir_flags*)

EOR

prototipo de la funcion EOR que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion EOR

5.7.2.7. void flag_C (uint32_t *Rn*, uint32_t *Rm*, uint32_t *Rd*, char * *dir_flag_C*)

flag_C

prototipo de la funcion flag_C que va a operar el registro de la bandera C

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>Rd</i>	operando ingresado
<i>dir_flag_C</i>	puntero que tiene la direccion de las banderas

5.7.2.8. void flag_N (uint32_t *Rd*, char * *dir_flag_N*)

flag_N

prototipo de la funcion flag_N que va a operar el registro de la bandera N

Parámetros

<i>Rd</i>	operando ingresado
<i>dir_flag_N</i>	puntero que tiene la direccion de las banderas

5.7.2.9. void flag_V (uint32_t *Rn*, uint32_t *Rm*, uint32_t *Rd*, char * *dir_flag_V*)

flag_V

prototipo de la funcion flag_V que va a operar el registro de la bandera V

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>Rd</i>	operando ingresado
<i>dir_flag_V</i>	puntero que tiene la direccion de las banderas

5.7.2.10. void flag_Z (uint32_t *Rd*, char * *dir_flag_Z*)

flag_Z

prototipo de la funcion flag_Z que va a operar el registro de la bandera Z

Parámetros

<i>Rd</i>	operando ingresado
<i>dir_flag_Z</i>	puntero que tiene la direccion de las banderas

5.7.2.11. void flags (uint32_t *Rn*, uint32_t *Rm*, uint32_t *Rd*, char * *dir_flags*)

flags

prototipo de la funcion flags que va a operar los registros de las banderas

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>Rd</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.7.2.12. uint32_t MOV (uint32_t *Rn*, char * *dir_flags*)

MOV

prototipo de la funcion MOV que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion MOV

5.7.2.13. `uint32_t MUL (uint32_t Rn, uint32_t Rm, char * dir_flags)`

MUL

prototipo de la funcion MUL que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion de las MUL

5.7.2.14. `void NOP (uint32_t * dir_reg)`

NOP

prototipo de la funcion NOP que va a operar los registros

Parámetros

<i>dir_reg</i>	puntero que tiene la direccion de los registros
----------------	-------------------------------------------------

5.7.2.15. `uint32_t ORR (uint32_t Rn, uint32_t Rm, char * dir_flags)`

ORR

prototipo de la funcion ORR que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion ORR

5.7.2.16. `uint32_t SBC (uint32_t Rn, uint32_t Rm, char * dir_flags)`

SBC

prototipo de la funcion SBC que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresaaado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la SBC

5.7.2.17. `uint32_t SUB (uint32_t Rn, uint32_t Rm, char * dir_flags)`

SUB

prototipo de la funcion SUB que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la SUB

5.7.2.18. void TST (uint32_t Rn, uint32_t Rm, char * dir_flags)

TST

prototipo de la funcion TST que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.8. Referencia del Archivo instrucciones.h

```
#include <stdint.h>
#include <stdio.h>
#include <stdlib.h>
```

Funciones

- void **flags** (uint32_t Rn, uint32_t Rm, uint32_t Rd, char *dir_flags)
flags

prototipo de la funcion flags que va a operar los registros de las banderas

- void **flag_N** (uint32_t Rd, char *dir_flag_N)
flag_N

prototipo de la funcion flag_N que va a operar el registro de la bandera N

- void **flag_Z** (uint32_t Rd, char *dir_flag_Z)
flag_Z

prototipo de la funcion flag_Z que va a operar el registro de la bandera Z

- void **flag_C** (uint32_t Rn, uint32_t Rm, uint32_t Rd, char *dir_flag_C)
flag_C

prototipo de la funcion flag_C que va a operar el registro de la bandera C

- void **flag_V** (uint32_t Rn, uint32_t Rm, uint32_t Rd, char *dir_flag_V)
flag_V

prototipo de la funcion flag_V que va a operar el registro de la bandera V

- void **CMN** (uint32_t Rn, uint32_t Rm, char *dir_flags)

CMN

prototipo de la funcion CMN que va a operar los registros

- void **CMP** (uint32_t Rn, uint32_t Rm, char *dir_flags)

CMP

prototipo de la funcion CMP que va a operar los registros

- uint32_t **MUL** (uint32_t Rn, uint32_t Rm, char *dir_flags)

MUL

prototipo de la funcion MUL que va a operar los registros

- void **TST** (uint32_t Rn, uint32_t Rm, char *dir_flags)

TST

prototipo de la funcion TST que va a operar los registros

- uint32_t **ADD** (uint32_t Rn, uint32_t Rm, char *dir_flags)

ADD

prototipo de la funcion ADD que va a operar los registros

- uint32_t **ADC** (uint32_t Rn, uint32_t Rm, char *dir_flags)

ADC

prototipo de la funcion ADC que va a operar los registros

- uint32_t **SUB** (uint32_t Rn, uint32_t Rm, char *dir_flags)

SUB

prototipo de la funcion SUB que va a operar los registros

- uint32_t **SBC** (uint32_t Rn, uint32_t Rm, char *dir_flags)

SBC

prototipo de la funcion SBC que va a operar los registros

- uint32_t **MOV** (uint32_t Rn, char *dir_flags)

MOV

prototipo de la funcion MOV que va a operar los registros

- uint32_t **AND** (uint32_t Rn, uint32_t Rm, char *dir_flags)

AND

prototipo de la funcion AND que va a operar los registros

- uint32_t **EOR** (uint32_t Rn, uint32_t Rm, char *dir_flags)

EOR

prototipo de la funcion EOR que va a operar los registros

- uint32_t **ORR** (uint32_t Rn, uint32_t Rm, char *dir_flags)

ORR

prototipo de la funcion ORR que va a operar los registros

- void **NOP** (uint32_t *dir_reg)

NOP

prototipo de la funcion NOP que va a operar los registros

5.8.1. Documentación de las funciones**5.8.1.1. uint32_t ADC (uint32_t Rn, uint32_t Rm, char * dir_flags)****ADC**

prototipo de la funcion ADC que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresaaado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la ADC

5.8.1.2. uint32_t ADD (uint32_t Rn, uint32_t Rm, char * dir_flags)**ADD**

prototipo de la funcion ADD que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresaaado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la ADD

5.8.1.3. `uint32_t AND (uint32_t Rn, uint32_t Rm, char * dir_flags)`

AND

prototipo de la funcion AND que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion AND

5.8.1.4. `void CMN (uint32_t Rn, uint32_t Rm, char * dir_flags)`

CMN

prototipo de la funcion CMN que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.8.1.5. `void CMP (uint32_t Rn, uint32_t Rm, char * dir_flags)`

CMP

prototipo de la funcion CMP que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.8.1.6. `uint32_t EOR (uint32_t Rn, uint32_t Rm, char * dir_flags)`

EOR

prototipo de la funcion EOR que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion EOR

5.8.1.7. `void flag_C (uint32_t Rn, uint32_t Rm, uint32_t Rd, char * dir_flag_C)`

flag_C

prototipo de la funcion flag_C que va a operar el registro de la bandera C

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>Rd</i>	operando ingresado
<i>dir_flag_C</i>	puntero que tiene la direccion de las banderas

5.8.1.8. `void flag_N (uint32_t Rd, char * dir_flag_N)`

flag_N

prototipo de la funcion `flag_N` que va a operar el registro de la bandera N

Parámetros

<i>Rd</i>	operando ingresado
<i>dir_flag_N</i>	puntero que tiene la direccion de las banderas

5.8.1.9. `void flag_V (uint32_t Rn, uint32_t Rm, uint32_t Rd, char * dir_flag_V)`

flag_V

prototipo de la funcion `flag_V` que va a operar el registro de la bandera V

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>Rd</i>	operando ingresado
<i>dir_flag_V</i>	puntero que tiene la direccion de las banderas

5.8.1.10. `void flag_Z (uint32_t Rd, char * dir_flag_Z)`

flag_Z

prototipo de la funcion `flag_Z` que va a operar el registro de la bandera Z

Parámetros

<i>Rd</i>	operando ingresado
<i>dir_flag_Z</i>	puntero que tiene la direccion de las banderas

5.8.1.11. `void flags (uint32_t Rn, uint32_t Rm, uint32_t Rd, char * dir_flags)`

flags

prototipo de la funcion flags que va a operar los registros de las banderas

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>Rd</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.8.1.12. uint32_t MOV (uint32_t *Rn*, char * *dir_flags*)**MOV**

prototipo de la funcion MOV que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion MOV

5.8.1.13. uint32_t MUL (uint32_t *Rn*, uint32_t *Rm*, char * *dir_flags*)**MUL**

prototipo de la funcion MUL que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion de las MUL

5.8.1.14. void NOP (uint32_t * dir_reg)

NOP

prototipo de la funcion NOP que va a operar los registros

Parámetros

<i>dir_reg</i>	puntero que tiene la direccion de los registros
----------------	-------------------------------------------------

5.8.1.15. uint32_t ORR (uint32_t Rn, uint32_t Rm, char * dir_flags)

ORR

prototipo de la funcion ORR que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la operacion ORR

5.8.1.16. uint32_t SBC (uint32_t Rn, uint32_t Rm, char * dir_flags)

SBC

prototipo de la funcion SBC que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresaaado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la SBC

5.8.1.17. `uint32_t SUB (uint32_t Rn, uint32_t Rm, char * dir_flags)`

SUB

prototipo de la funcion SUB que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

Devuelve

retorna el resultado de la SUB

5.8.1.18. `void TST (uint32_t Rn, uint32_t Rm, char * dir_flags)`

TST

prototipo de la funcion TST que va a operar los registros

Parámetros

<i>Rn</i>	operando ingresado
<i>Rm</i>	operando ingresado
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.9. Referencia del Archivo main.c

```
#include <curses.h>
#include "imprimir.h"
#include "instrucciones.h"
#include "instruc_desplazamiento.h"
#include "salto.h"
#include "decoder.h"
#include <time.h>
```

'defines'

- `#define N 0`

- #define Z 1
- #define C 2
- #define V 3
- #define LR 14
- #define PC 15

Funciones

- int main (void)

5.9.1. Documentación de los 'defines'

5.9.1.1. #define C 2

5.9.1.2. #define LR 14

5.9.1.3. #define N 0

5.9.1.4. #define PC 15

5.9.1.5. #define V 3

5.9.1.6. #define Z 1

5.9.2. Documentación de las funciones

5.9.2.1. int main (void)

5.10. Referencia del Archivo README.md

5.11. Referencia del Archivo salto.c

```
#include "salto.h"
```

'defines'

- #define N 0
- #define Z 1
- #define C 2
- #define V 3
- #define LR 14
- #define PC 15

Funciones

- void B (uint32_t label, uint32_t *dir_reg)

B

prototipo de la funcion B que va a modificar el registro PC

- void **BL** (uint32_t label, uint32_t *dir_reg)

BL

prototipo de la funcion BL que va a modificar el registro PC

- void **BX** (uint32_t *dir_reg)

BX

prototipo de la funcion BX que va a modificar el registro PC

- void **BEQ** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BEQ

prototipo de la funcion BEQ que va a modificar el registro PC

- void **BNE** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BNE

prototipo de la funcion BNE que va a modificar el registro PC

- void **BCS** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BCS

prototipo de la funcion BCS que va a modificar el registro PC

- void **BCC** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BCC

prototipo de la funcion BCC que va a modificar el registro PC

- void **BMI** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BMI

prototipo de la funcion BMI que va a modificar el registro PC

- void **BPL** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BPL

prototipo de la funcion BPL que va a modificar el registro PC

- void **BVS** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BVS

prototipo de la funcion BVS que va a modificar el registro PC

- void **BVC** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BVC

prototipo de la funcion BVC que va a modificar el registro PC

- void **BHI** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BHI

prototipo de la funcion BHI que va a modificar el registro PC

- void **BLS** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BLS

prototipo de la funcion BLS que va a modificar el registro PC

- void **BGE** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BGE

prototipo de la funcion BGE que va a modificar el registro PC

- void **BLT** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BLT

prototipo de la funcion BLT que va a modificar el registro PC

- void **BGT** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BGT

prototipo de la funcion BGT que va a modificar el registro PC

- void **BLE** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BLE

prototipo de la funcion BLE que va a modificar el registro PC

- void **BAL** (uint32_t label, uint32_t *dir_reg)

BAL

prototipo de la funcion BAL que va a modificar el registro PC

5.11.1. Documentación de los 'defines'

5.11.1.1. #define C 2

5.11.1.2. #define LR 14

5.11.1.3. #define N 0

5.11.1.4. `#define PC 15`

5.11.1.5. `#define V 3`

5.11.1.6. `#define Z 1`

5.11.2. Documentación de las funciones

5.11.2.1. `void B (uint32_t label, uint32_t * dir_reg)`

B

prototipo de la funcion B que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros

5.11.2.2. `void BAL (uint32_t label, uint32_t * dir_reg)`

BAL

prototipo de la funcion BAL que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros

5.11.2.3. `void BCC (uint32_t label, uint32_t * dir_reg, char * dir_flags)`

BCC

prototipo de la funcion BCC que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.4. void BCS (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BCS

prototipo de la funcion BCS que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.5. void BEQ (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BEQ

prototipo de la funcion BEQ que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.6. void BGE (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BGE

prototipo de la funcion BGE que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.7. void BGT (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BGT

prototipo de la funcion BGT que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.8. void BHI (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BHI

prototipo de la funcion BHI que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.9. void BL (uint32_t *label*, uint32_t * *dir_reg*)

BL

prototipo de la funcion BL que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros

5.11.2.10. void BLE (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BLE

prototipo de la funcion BLE que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.11. void BLS (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BLS

prototipo de la funcion BLS que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.12. void BLT (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BLT

prototipo de la funcion BLT que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.13. void BMI (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BMI

prototipo de la funcion BMI que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.14. void BNE (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BNE

prototipo de la funcion BNE que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.15. void BPL (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BPL

prototipo de la funcion BPL que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.16. void BVC (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BVC

prototipo de la funcion BVC que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.17. void BVS (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BVS

prototipo de la funcion BVS que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.11.2.18. void BX (uint32_t * *dir_reg*)

BX

prototipo de la funcion BX que va a modificar el registro PC

Parámetros

<i>dir_reg</i>	puntero que tiene la direccion de los registros
----------------	-------------------------------------------------

5.12. Referencia del Archivo salto.h

```
#include <stdint.h>
#include <stdio.h>
#include <stdlib.h>
```

Funciones

- void **B** (uint32_t label, uint32_t *dir_reg)

B

prototipo de la funcion B que va a modificar el registro PC

- void **BL** (uint32_t label, uint32_t *dir_reg)

BL

prototipo de la funcion BL que va a modificar el registro PC

- void **BX** (uint32_t *dir_reg)

BX

prototipo de la funcion BX que va a modificar el registro PC

- void **BEQ** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BEQ

prototipo de la funcion BEQ que va a modificar el registro PC

- void **BNE** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BNE

prototipo de la funcion BNE que va a modificar el registro PC

- void **BCS** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BCS

prototipo de la funcion BCS que va a modificar el registro PC

- void **BCC** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BCC

prototipo de la funcion BCC que va a modificar el registro PC

- void **BMI** (uint32_t label, uint32_t *dir_reg, char *dir_flags)
BMI

prototipo de la funcion BMI que va a modificar el registro PC

- void **BPL** (uint32_t label, uint32_t *dir_reg, char *dir_flags)
BPL

prototipo de la funcion BPL que va a modificar el registro PC

- void **BVS** (uint32_t label, uint32_t *dir_reg, char *dir_flags)
BVS

prototipo de la funcion BVS que va a modificar el registro PC

- void **BVC** (uint32_t label, uint32_t *dir_reg, char *dir_flags)
BVC

prototipo de la funcion BVC que va a modificar el registro PC

- void **BHI** (uint32_t label, uint32_t *dir_reg, char *dir_flags)
BHI

prototipo de la funcion BHI que va a modificar el registro PC

- void **BLS** (uint32_t label, uint32_t *dir_reg, char *dir_flags)
BLS

prototipo de la funcion BLS que va a modificar el registro PC

- void **BGE** (uint32_t label, uint32_t *dir_reg, char *dir_flags)
BGE

prototipo de la funcion BGE que va a modificar el registro PC

- void **BLT** (uint32_t label, uint32_t *dir_reg, char *dir_flags)
BLT

prototipo de la funcion BLT que va a modificar el registro PC

- void **BGT** (uint32_t label, uint32_t *dir_reg, char *dir_flags)
BGT

prototipo de la funcion BGT que va a modificar el registro PC

- void **BLE** (uint32_t label, uint32_t *dir_reg, char *dir_flags)

BLE

prototipo de la funcion BLE que va a modificar el registro PC

- void **BAL** (uint32_t label, uint32_t *dir_reg)

BAL

prototipo de la funcion BAL que va a modificar el registro PC

5.12.1. Documentación de las funciones

5.12.1.1. void B (uint32_t label, uint32_t * dir_reg)

B

prototipo de la funcion B que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros

5.12.1.2. void BAL (uint32_t label, uint32_t * dir_reg)

BAL

prototipo de la funcion BAL que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros

5.12.1.3. void BCC (uint32_t label, uint32_t * dir_reg, char * dir_flags)

BCC

prototipo de la funcion BCC que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.4. void BCS (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BCS

prototipo de la funcion BCS que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.5. void BEQ (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BEQ

prototipo de la funcion BEQ que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.6. void BGE (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BGE

prototipo de la funcion BGE que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.7. void BGT (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BGT

prototipo de la funcion BGT que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.8. void BHI (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BHI

prototipo de la funcion BHI que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.9. void BL (uint32_t *label*, uint32_t * *dir_reg*)

BL

prototipo de la funcion BL que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros

5.12.1.10. void BLE (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BLE

prototipo de la funcion BLE que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.11. void BLS (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BLS

prototipo de la funcion BLS que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.12. void BLT (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BLT

prototipo de la funcion BLT que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.13. void BMI (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BMI

prototipo de la funcion BMI que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.14. void BNE (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BNE

prototipo de la funcion BNE que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.15. void BPL (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BPL

prototipo de la funcion BPL que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.16. void BVC (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BVC

prototipo de la funcion BVC que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.17. void BVS (uint32_t *label*, uint32_t * *dir_reg*, char * *dir_flags*)

BVS

prototipo de la funcion BVS que va a modificar el registro PC

Parámetros

<i>label</i>	operando ingresado
<i>dir_reg</i>	puntero que tiene la direccion de los registros
<i>dir_flags</i>	puntero que tiene la direccion de las banderas

5.12.1.18. void BX (uint32_t * *dir_reg*)

BX

prototipo de la funcion BX que va a modificar el registro PC

Parámetros

<i>dir_reg</i>	puntero que tiene la direccion de los registros
----------------	-------------------------------------------------

5.13. Referencia del Archivo test.c

```
#include "imprimir.h"
#include "instrucciones.h"
#include "instruc_desplazamiento.h"
#include <curses.h>
```

'defines'

- #define N 0
- #define Z 1
- #define C 2
- #define V 3

Funciones

- int main ()

TEXT

Este archivo es para probar las librerías.

5.13.1. Documentación de los 'defines'

5.13.1.1. #define C 2

5.13.1.2. #define N 0

5.13.1.3. #define V 3

5.13.1.4. #define Z 1

5.13.2. Documentación de las funciones

5.13.2.1. int main (void)

TEXT

Este archivo es para probar las librerías.